MAC 5711 – Análise de Algoritmos
SEGUNDO SEMESTRE DE 2005
Primeira Prova – 13 de setembro

Nome do aluno: ____ **Curso**:____________

Assinatura: ____

No. USP: ______________________ **Professor**:________________________________

Instruções

1. Não destaque as folhas deste caderno.
2. A prova pode ser feita a lápis.
3. A legibilidade também faz parte da nota!
4. A prova consta de 4 questões. Verifique antes de começar a prova se o seu caderno de questões está completo.
5. Não é permitido o uso de folhas avulsas para rascunho.
6. Não é necessário apagar rascunhos no caderno de questão mas especifique qual é a resposta e qual é o rascunho.
7. A prova é sem consulta.

Não escrever nesta parte da folha

Questão	Nota	Observação
1		
2		
3		
4		
Total		

Boa Sorte !

1. [1,0 ponto]
   Seja $f : \mathbf{N} \to \mathbf{N}$ uma função assintoticamente positiva. Mostre que $O(O(f(n))) = O(f(n))$.

2. [2,5 pontos]
   Resolva a recorrência

$$T(n) = \begin{cases} 1 & \text{se } n = 1 \\ T(n/2) + n \lg n & \text{se } n = 2, 4, 8, 16, \ldots. \end{cases}$$

   definida apenas para n potência de 2. Resolva a recorrência dando uma fórmula fechada para $T(n)$ (quando n é potência de 2).

3. [3,5 pontos]
Considere uma seqüência de vetores de inteiros

$$A_k[1\,..\,2^k], A_{k-1}[1\,..\,2^{k-1}], \ldots, A_1[1\,..\,2^1], A_0[1\,..\,2^0].$$

Suponha que os inteiros armazenados nos vetores são dois a dois distintos e que cada vetor está em ordem crescente.

O seguinte problema foi proposto a uma classe de alunos: escreva um algoritmo que reuna, por meio de sucessivas operações de intercalação como as efetuadas pela rotina *intercala* do *mergesort*, o conteúdo dos vetores $A_0, \ldots, A_k$ em um único vetor crescente $B[1\,..\,n]$, onde $n = 2^{k+1} - 1$.

Entre várias, duas soluções para o problema, esquematizadas abaixo, apareceram:

Algoritmo JUNTA1($A_0, \ldots, A_k$)
1. $B[1\,..\,2^k] \leftarrow A_k[1\,..\,2^k]$
2. $m \leftarrow 2^k$
3. **para** $i \leftarrow k-1$ **até** 0 **faça**
4. $\quad B[1\,..\,m+2^i] \leftarrow \text{INTERCALA}(B[1\,..\,m], A_i[1\,..\,2^i])$
5. $\quad m \leftarrow m + 2^i$
6. **devolva** B

Algoritmo JUNTA2($A_0, \ldots, A_k$)
1. $B[1\,..\,1] \leftarrow A_0[1\,..\,1]$
2. $m \leftarrow 1$
3. **para** $i \leftarrow 1$ **até** k **faça**
4. $\quad B[1\,..\,m+2^i] \leftarrow \text{INTERCALA}(B[1\,..\,m], A_i[1\,..\,2^i])$
5. $\quad m \leftarrow m + 2^i$;
6. **devolva** B

Em termos de complexidade de tempo, as duas soluções são ou não são equivalentes? Se não são, qual das duas é melhor? Justifique a sua resposta deduzindo o comportamento assintótico exato (ou seja, com a notação Θ) da complexidade de tempo de cada uma, e comparando estes comportamentos. *Dica:* calcule antes de mais nada o valor de m no início de cada iteração do *para* da linha 4.

4. [3,0 pontos]
Considere o seguinte algoritmo que determina o segundo maior elemento de um vetor $v[1\,..\,n]$ com $n \geq 2$ números positivos distintos.

Algoritmo Máximo (v, n)

```
1.  maior ← 0
2.  segundo_maior ← 0
3.  para i ← 1 até n faça
4.     se v[i] > maior
5.        então segundo_maior ← maior
6.              maior ← v[i]
7.        senão se v[i] > segundo_maior
8.                 então segundo_maior ← v[i]
9.  devolva segundo_maior
```

Suponha que v é uma permutação de 1 a n escolhida ao acaso dentre todas as permutações de 1 a n, de acordo com a distribuição uniforme de probabilidade. Seja X o número de vezes que a variável *segundo_maior* é alterada (ou seja, o número de execuções das linhas 5 e 8 do algoritmo) numa chamada de Máximo(v, n). Note que X é uma variável aleatória. Calcule o valor esperado de X.